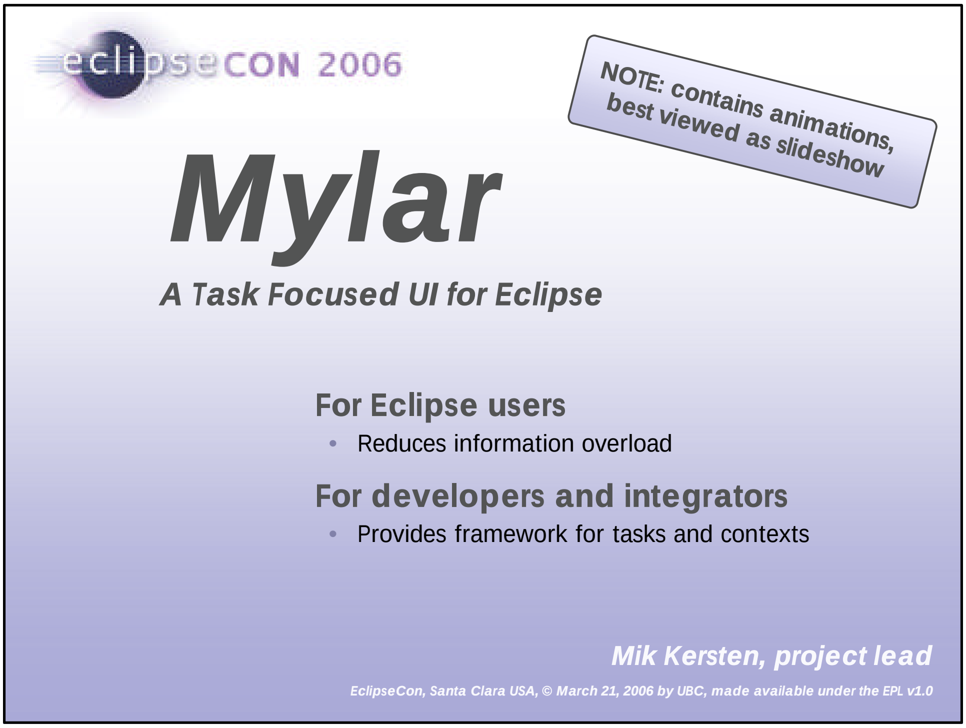


eclipseCON 2006

NOTE: contains animations, best viewed as slideshow

Mylar

A Task Focused UI for Eclipse

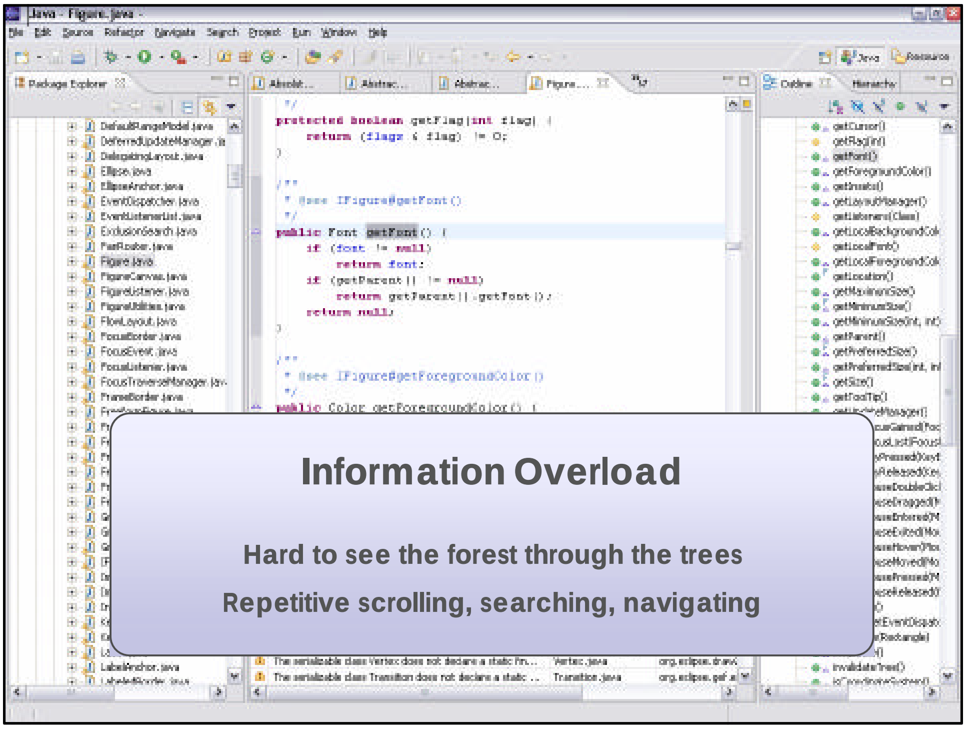
For Eclipse users

- Reduces information overload

For developers and integrators

- Provides framework for tasks and contexts

Mik Kersten, project lead



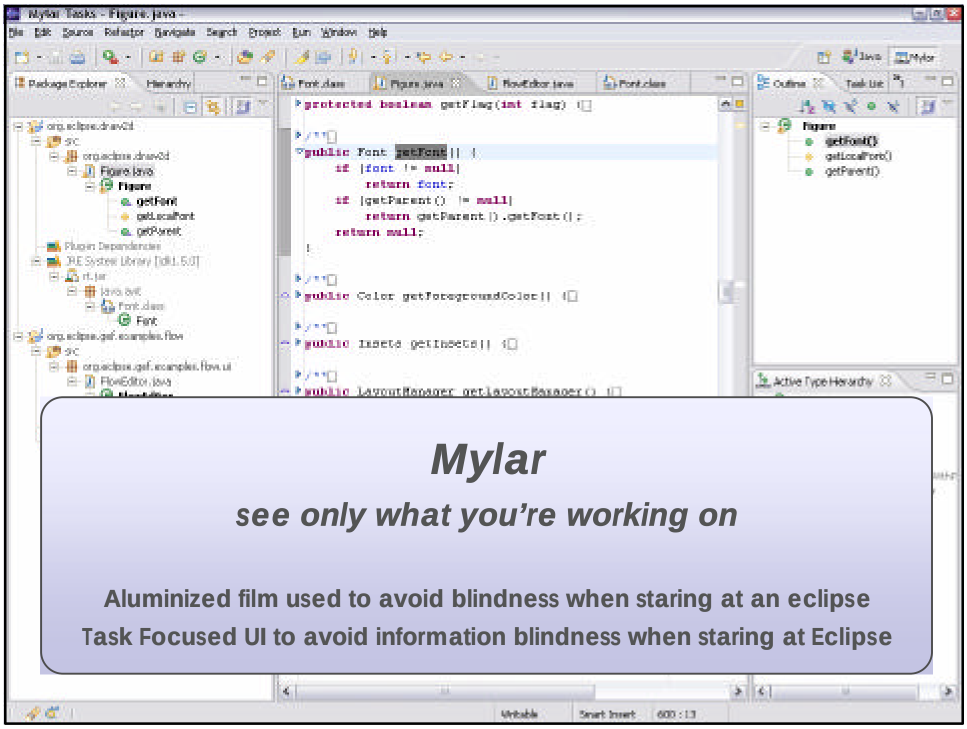
```
 */
protected boolean getFlag(int flag) {
    return (flags & flag) != 0;
}

/**
 * @see IFigure#getFont()
 */
public Font getFont() {
    if (font != null)
        return font;
    if (getParent() != null)
        return getParent().getFont();
    return null;
}

/**
 * @see IFigure#getForegroundColor()
 */
public Color getForegroundColor() {
```

```
The serializable class Vertex does not declare a static fin...   Vertex.java       org.eclipse.draw2
The serializable class Transition does not declare a static ...  Transition.java   org.eclipse.gef.e
```

Information Overload

Hard to see the forest through the trees

Repetitive scrolling, searching, navigating



```
protected boolean getFlag(int flag) {

/**
public Font getFont() {
    if (font != null)
        return font;
    if (getParent() != null)
        return getParent().getFont();
    return null;
}

/**
public Color getForegroundColor() {

/**
public Insets getInsets() {

/**
public LayoutManager getLayoutManager() {
```

Mylar

see only what you're working on

Aluminized film used to avoid blindness when staring at an eclipse
Task Focused UI to avoid information blindness when staring at Eclipse

Overview

Working with Mylar

- Tasks and contexts make working with large systems easier
- Bugs/issues/tasks are integrated and easy to manage

Demos highlighting key features

- Working with task context
- Eclipse integration (SDK)
- Repository integration (Bugzilla, JIRA)

Building on Mylar

- Internals & architecture
- Framework & APIs

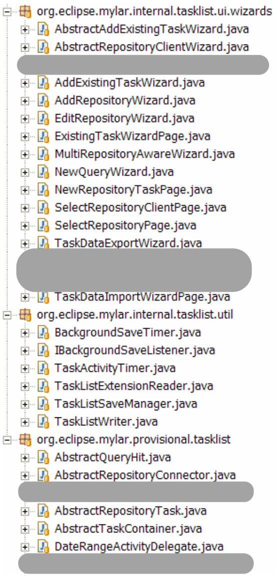
Demo 1: task context

Without Mylar

- Manually manage context
- Use working sets, filters

With Mylar

- Indicate what task you're working on
- Programming activity forms context for that task
- Context becomes explicit in the UI

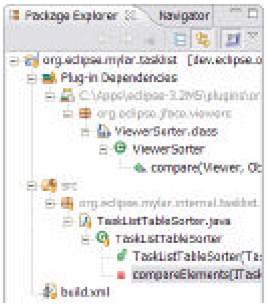


```
org.eclipse.mylar.internal.tasklist.ui.wizards
    AbstractAddExistingTaskWizard.java
    AbstractRepositoryClientWizard.java

    AddExistingTaskWizard.java
    AddRepositoryWizard.java
    EditRepositoryWizard.java
    ExistingTaskWizardPage.java
    MultiRepositoryAwareWizard.java
    NewQueryWizard.java
    NewRepositoryTaskPage.java
    SelectRepositoryClientPage.java
    SelectRepositoryPage.java
    TaskDataExportWizard.java

    TaskDataImportWizardPage.java
org.eclipse.mylar.internal.tasklist.util
    BackgroundSaveTimer.java
    IBackgroundSaveListener.java
    TaskActivityTimer.java
    TaskListExtensionReader.java
    TaskListSaveManager.java
    TaskListWriter.java
org.eclipse.mylar.provisional.tasklist
    AbstractQueryHit.java
    AbstractRepositoryConnector.java

    AbstractRepositoryTask.java
    AbstractTaskContainer.java
    DateRangeActivityDelegate.java
```

Task context

Tasks

- User-defined unit of work, e.g. bug report



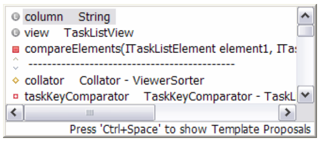
```
org.eclipse.mylar.tasklist [dev.eclipse.o
    Plug-in Dependencies
        C:\Apps\eclipse-3.2M5\plugins\or
            org.eclipse.jface.viewers
                ViewerSorter.class
                    ViewerSorter
                        compare(Viewer, Ob
    src
        org.eclipse.mylar.internal.tasklist.
            TaskListTableSorter.java
                TaskListTableSorter
                    TaskListTableSorter(Ta
                    compareElements(ITask
    build.xml
```

Context

- Mylar monitors your interaction
- Creates degree-of-interest model
- What you touch is in your context
- Actively managed as you work
- Stored and easy to recall

Focused UI

- Views: filtering, decoration
- Editors: folding, content assist
- Context switching, editor management



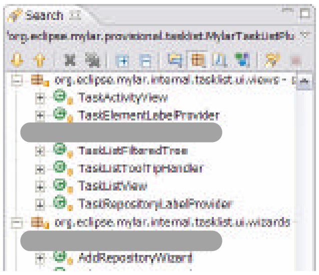
```
column    String
view    TaskListView
compareElements(ITaskListElement element1, ITa
 ----------------------------------------
collator    Collator - ViewerSorter
taskKeyComparator    TaskKeyComparator - TaskL
```

Press 'Ctrl+Space' to show Template Proposals

Demo 2: integration

Search

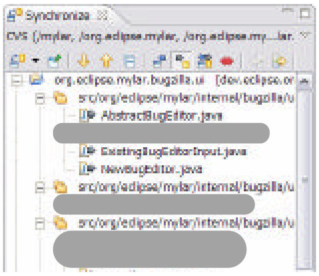
- Repeatedly search and scanning results to find what's related



```
Search
'org.eclipse.mylar.provisional.tasklist.MylarTaskListPlu
org.eclipse.mylar.internal.tasklist.ui.views
    TaskActivityView
    TaskElementLabelProvider
    TaskListFilteredTree
    TaskListToolTipHandler
    TaskListView
    TaskRepositoryLabelProvider
org.eclipse.mylar.internal.tasklist.ui.wizards
    AddRepositoryWizard
```

Synchronize

- Sometimes you only want to commit or update a subset

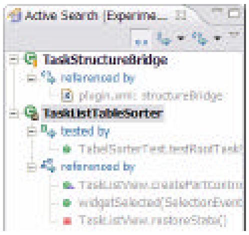


```
Synchronize
CVS (/mylar, /org.eclipse.mylar, /org.eclipse.my....lar.
org.eclipse.mylar.bugzilla.ui   [dev.eclipse.or
    src/org/eclipse/mylar/internal/bugzilla/u
        AbstractBugEditor.java
        ExistingBugEditorInput.java
        NewBugEditor.java
    src/org/eclipse/mylar/internal/bugzilla/u
    src/org/eclipse/mylar/internal/bugzilla/u
```

Integration

Active search

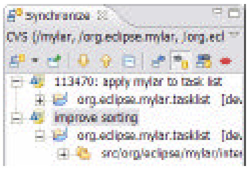
- Related elements become interesting
- Search is seeded and scoped by task context



```
Active Search [Experime...
TaskStructureBridge
    referenced by
        plugin.xml: structureBridge
TaskListTableSorter
    tested by
        TableSorterTest.testRootTask
    referenced by
        TaskListView.createPartContro
        widgetSelected(SelectionEvent
        TaskListView.restoreState()
```

Active change sets

- Commit just a piece of what you've changed
- Automatically managed with tasks
- Resources in context mapped to change set



```
Synchronize
CVS (/mylar, /org.eclipse.mylar, /org.ecl
113470: apply mylar to task list
    org.eclipse.mylar.tasklist   [de
improve sorting
    org.eclipse.mylar.tasklist   [de
        src/org/eclipse/mylar/inter
```

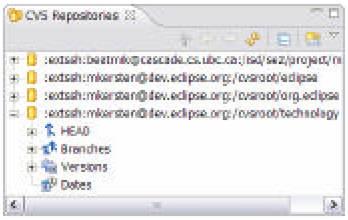
Demo 3: task repositories

Without Mylar

- Work with various web UIs to manage bugs/issues/tasks

With Mylar

- Task management is integrated
- Similar to source repositories
- Get persistence, offline editing


CVS Repositories

```
+ :extssh:beatmik@cascade.cs.ubc.ca:/isd/sez/project/m
+ :extssh:mkersten@dev.eclipse.org:/cvsroot/eclipse
+ :extssh:mkersten@dev.eclipse.org:/cvsroot/org.eclipse
- :extssh:mkersten@dev.eclipse.org:/cvsroot/technology
    + HEAD
    + Branches
    + Versions
      Dates
```

Task repositories

Connectors

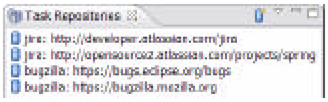
- Similar to source repositories
- Support Bugzilla and JIRA

Tasks

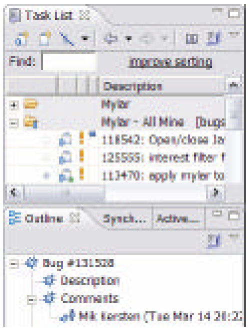
- Local
- Web linked
- Repository queries
- Authoring, offline editing

One integrated task list

- Personalized notes, reminders
- Archive, filters, notifications


Task Repositories

```
jira: http://developer.atlassian.com/jira
jira: http://opensource2.atlassian.com/projects/spring
bugzilla: https://bugs.eclipse.org/bugs
bugzilla: https://bugzilla.mozilla.org
```

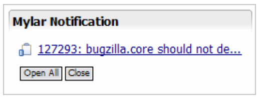

Task List

Find: improve sorting

```
Description
+ Mylar
- Mylar - All Mine   [bugs
      118542: Open/close Ja
      125555: interest filter f
      113470: apply mylar to
```

Outline | Synch... | Active...

```
- Bug #131528
    Description
  - Comments
      Mik Kersten (Tue Mar 14 20:2
```

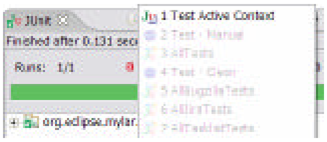

Mylar Notification

127293: bugzilla.core should not de...

There's more…

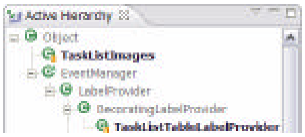
Automation

- Testing context via Active Test Suite



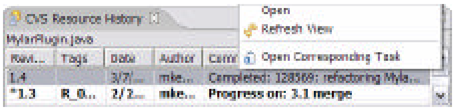
Context views

- Active Type Hierarchy



Everything is linked

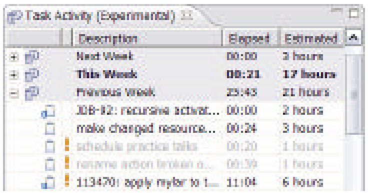
- Tasks to context to resources



Revi...	Tags	Date	Author	Comm
1.4		3/7/...	mke...	Completed: 128569: refactoring Myla...
*1.3	R_0...	2/2...	mke...	Progress on: 3.1 merge

Easy to build on

- XP planning via Task Activity View



Description	Elapsed	Estimated
Next Week	00:00	3 hours
This Week	00:21	17 hours
Previous Week	23:43	21 hours
108-82: recursive activat...	00:00	2 hours
make changed resource...	00:24	3 hours
schedule practice talks	00:20	1 hours
rename action broken o...	00:39	1 hours
113470: apply mylar to t...	11:04	6 hours

Changing how we work with Eclipse

What you need to do

- Buy into working with tasks
- Have patience with an evolving UI (v0.5)

Once tasks are explicit

- Reduces information overload
- Information you need to get work done is at your finger tips
- Context switching and recalling old tasks become effortless
- UI automation (e.g. working sets, search, commit messages)
- UI for task management is consistent and integrated
- Keeps you in Eclipse and out of your browser and inbox

under the hood

Mylar's context model

Interaction history

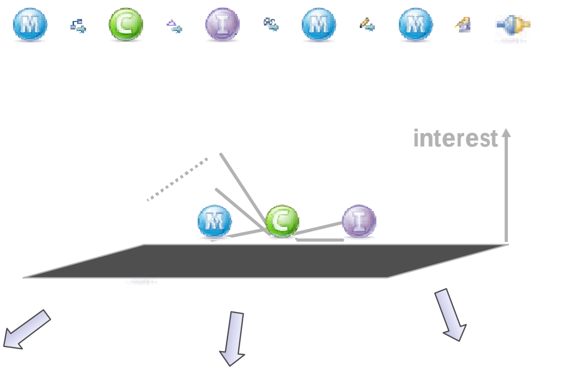
- InteractionEvent stream
- Origin, handle, type, date

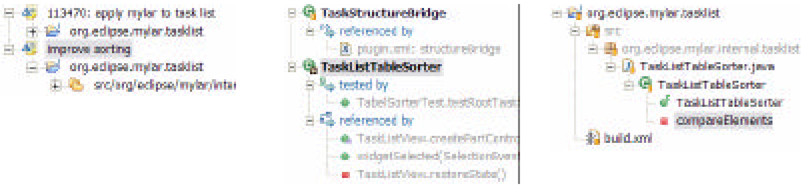
Context (Core)

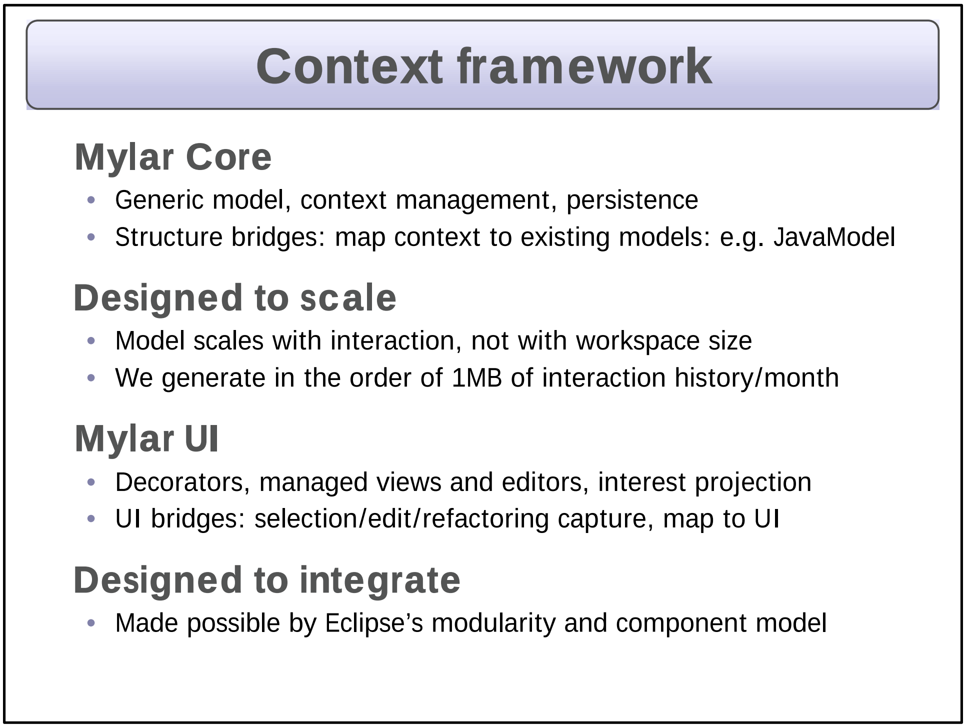
- Degree-of-interest graph
- Degree-of-separation scope
- Scaling factors

Projections (UI)

- Views, editors, files




113470: apply mylar to task list
org.eclipse.mylar.tasklist
improve sorting
org.eclipse.mylar.tasklist
src/org/eclipse/mylar/inte
TaskStructureBridge
referenced by
plugin.xml: structureBridge
TaskListTableSorter
tested by
TabelSorterTest.testRootTask
referenced by
TaskListView.createPartContro
widgetSelected(SelectionEvent
TaskListView.restoreState()
org.eclipse.mylar.tasklist
src
org.eclipse.mylar.internal.tasklist
TaskListTableSorter.java
TaskListTableSorter
TaskListTableSorter
compareElements
build.xml



Context framework

Mylar Core

- Generic model, context management, persistence
- Structure bridges: map context to existing models: e.g. JavaModel

Designed to scale

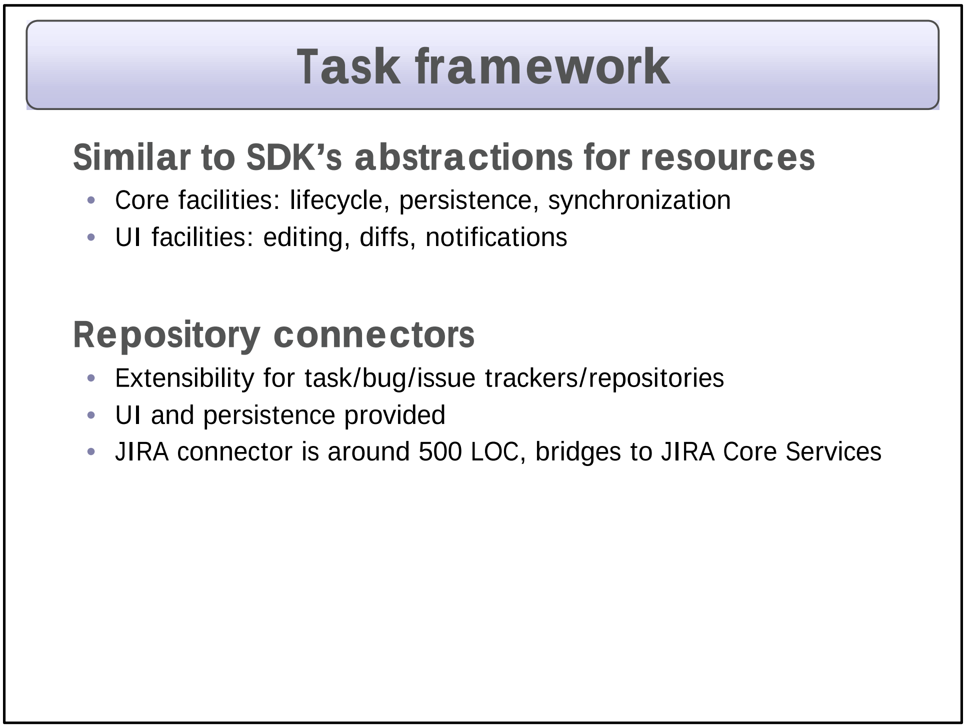
- Model scales with interaction, not with workspace size
- We generate in the order of 1MB of interaction history/month

Mylar UI

- Decorators, managed views and editors, interest projection
- UI bridges: selection/edit/refactoring capture, map to UI

Designed to integrate

- Made possible by Eclipse's modularity and component model



Task framework

Similar to SDK's abstractions for resources

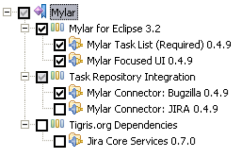
- Core facilities: lifecycle, persistence, synchronization
- UI facilities: editing, diffs, notifications

Repository connectors

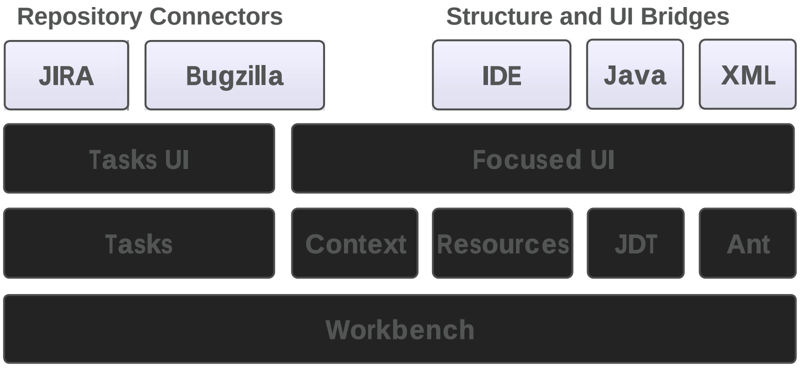
- Extensibility for task/bug/issue trackers/repositories
- UI and persistence provided
- JIRA connector is around 500 LOC, bridges to JIRA Core Services

Extensibility

Make tasks a first class abstraction, connected to resources with context



- ▣ Mylar
  - ☑ Mylar for Eclipse 3.2
    - ☑ Mylar Task List (Required) 0.4.9
    - ☑ Mylar Focused UI 0.4.9
  - ▣ Task Repository Integration
    - ☑ Mylar Connector: Bugzilla 0.4.9
    - □ Mylar Connector: JIRA 0.4.9
  - □ Tigris.org Dependencies
    - □ Jira Core Services 0.7.0



Repository Connectors

JIRA

Bugzilla

Structure and UI Bridges

IDE

Java

XML

Tasks UI

Focused UI

Tasks

Context

Resources

JDT

Ant

Workbench

Provisional APIs

Tasks API

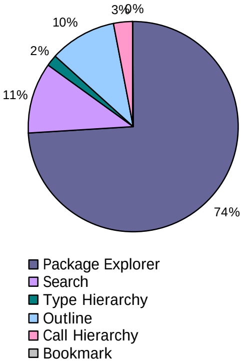
- Currently supports two connectors: Bugzilla, JIRA
- Extensible to other kinds of trackers and repositories
- Ready for early API adopters to influence and stabilize

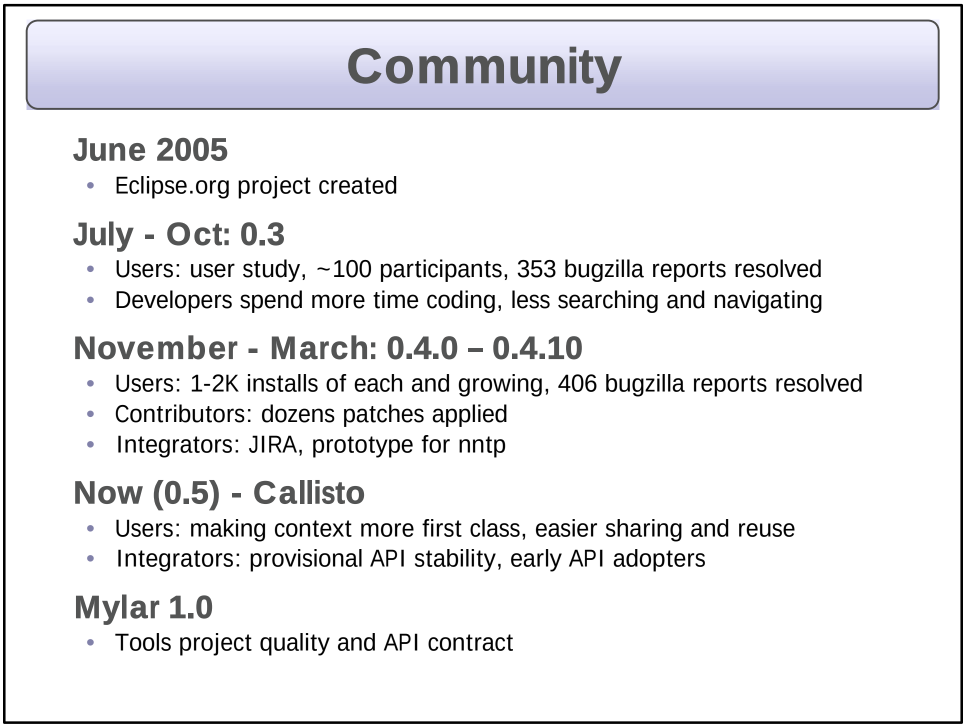
Context API

- Currently supports: Resources, JDT, PDE, Ant
- Has been stable, but revision coming

Monitor API

- Usage monitoring based on interaction history
- Studied ~100 developers voluntarily using Mylar
- Privacy, obfuscation, reporting





Community

June 2005

- Eclipse.org project created

July - Oct: 0.3

- Users: user study, ~100 participants, 353 bugzilla reports resolved
- Developers spend more time coding, less searching and navigating

November - March: 0.4.0 – 0.4.10

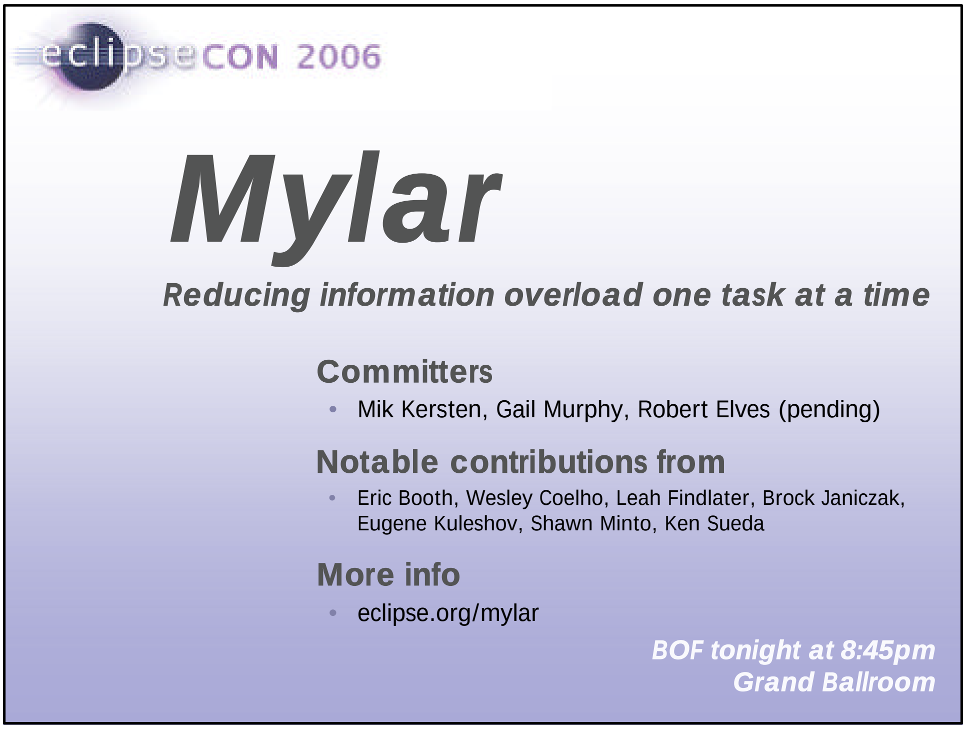
- Users: 1-2K installs of each and growing, 406 bugzilla reports resolved
- Contributors: dozens patches applied
- Integrators: JIRA, prototype for nntp

Now (0.5) - Callisto

- Users: making context more first class, easier sharing and reuse
- Integrators: provisional API stability, early API adopters

Mylar 1.0

- Tools project quality and API contract



eclipseCON 2006

Mylar

Reducing information overload one task at a time

Committers

- Mik Kersten, Gail Murphy, Robert Elves (pending)

Notable contributions from

- Eric Booth, Wesley Coelho, Leah Findlater, Brock Janiczak, Eugene Kuleshov, Shawn Minto, Ken Sueda

More info

- eclipse.org/mylar

BOF tonight at 8:45pm
Grand Ballroom